Integrating Structural Semantic Knowledge for Enhanced Information Extraction Pre-training

Xiaoyang Yi[1,3,4,*], Yuru Bao[1,3,4,*], Jian Zhang[1,2,3,4,†], Yifang Qin[2,3,4], Faxin Lin[1,3,4]
[1]College of Cyber Science, Nankai University
[2]College of Computer Science, Nankai University
[3]Tianjin Key Laboratory of Network and Data Security Technology
[4]Key Laboratory of Data and Intelligent System Security Ministry of Education
[†]**Correspondence:** zhang.jian@nankai.edu.cn

Abstract

Information Extraction (IE), aiming to extract structured information from unstructured natural language texts, can significantly benefit from pre-trained language models. However, existing pre-training methods solely focus on exploiting the textual knowledge, relying extensively on annotated large-scale datasets, which is labor-intensive and thus limits the scalability and versatility of the resulting models. To address these issues, we propose SKIE, a novel pre-training framework tailored for IE that integrates structural semantic knowledge via contrastive learning, effectively alleviating the annotation burden. Specifically, SKIE utilizes Abstract Meaning Representation (AMR) as a low-cost supervision source to boost model performance without human intervention. By enhancing the topology of AMR graphs, SKIE derives high-quality cohesive subgraphs as additional training samples, providing diverse multi-level structural semantic knowledge. Furthermore, SKIE refines the graph encoder to better capture cohesive information and edge relation information, thereby improving the pre-training efficacy. Extensive experimental results demonstrate that SKIE outperforms state-of-the-art baselines across multiple IE tasks and showcases exceptional performance in few-shot and zero-shot settings.

1 Introduction

Information Extraction (IE) aims to extract structured information from unstructured natural language texts (Grishman and Sundheim, 1996; Grishman, 2019), which encompasses several subtasks such as Named Entity Recognition (NER) (Shen et al., 2023; Ghosh et al., 2023), Relation Extraction (RE) (Sun et al., 2023; Wu et al., 2023), and Event Extraction (EE) (Guzman Nateras et al., 2023; Liu et al., 2023). Considering the inherent connections among these subtasks, recent methods propose to jointly resolve them within a unified framework, capitalizing on the generalization versatility of pre-trained language models (PLMs).

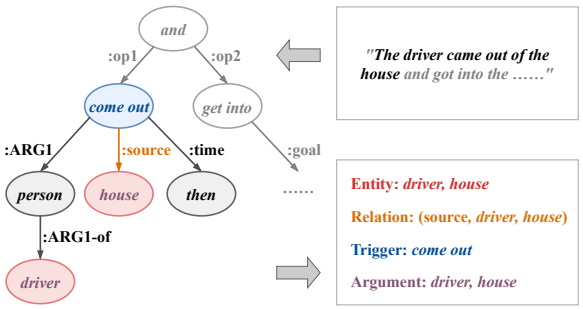

Figure 1: An example from the WikiEvents dataset. The AMR graph (left) highlights key elements: the entities *driver* and *house* are connected by the relation "source". The event trigger is *come out*, with the entities themselves serving as the arguments for the event.

For instance, UIE (Lu et al., 2022) embeds schema-based prompts into the corpora to pre-train a text-to-structure generative PLM, enabling it to generate uniform representations. USM (Lou et al., 2023) utilizes three kinds of supervised datasets and employs unified token linking to structure information during pre-training. Mirror (Zhu et al., 2023) designs a unified data interface to reorganize datasets into multi-slot tuples for pre-training. MetaRetriever (Cong et al., 2023) introduces a Meta-Pretraining Algorithm to retrieve task-specific knowledge from PLMs for IE tasks.

However, existing pre-training methods suffer from two major challenges. First, the high cost of annotation restricts existing datasets for IE tasks to a few predefined categories and small data volumes (Lou et al., 2023), limiting the amount of supervised datasets available for pre-training. Second, these methods are constrained to solely utilizing annotated textual knowledge, neglecting the potential structural semantic knowledge inherent in texts, which hinders their ability to leverage complex structural knowledge.

[*]The same contribution

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 2156–2171
November 12-16, 2024

A feasible solution is to generate self-supervised signals from extensive unsupervised data by leveraging their structural semantic knowledge, instead of relying on limited supervised data. Abstract Meaning Representation (AMR) (Banarescu et al., 2013), which has demonstrated its ability to capture structural semantic knowledge within texts without additional human effort (Bai et al., 2022; Wang et al., 2015), stands out as a fitting choice. Figure 1 illustrates an example of a text segment and its corresponding AMR graph. The text is converted into an AMR graph through AMR parsing, where nodes represent basic semantic units such as entities and predicates, while edges denote their semantic relations (Bai et al., 2022).

Armed with this insight, we propose SKIE, a novel pre-training method that integrates **S**tructural semantic **K**nowledge to enhance the model's versatility across multiple **IE** tasks. SKIE leverages AMR parsing to generate self-supervised signals, offering a flexible and general approach to semantic representations. To capture more diverse semantic structures, SKIE introduces cohesive subgraphs, which are densely connected subsets of pivotal nodes within the graph. Then, contrastive learning is employed to bridge the associations between texts and graphs.

Specifically, SKIE comprises three key modules: the *topology enhancement* module, the *encoding cohesion* module, and the *contrastive learning* module. Capitalizing on the cohesion-guided topology enhancement, we extract cohesive subgraphs from AMR graphs to acquire diverse multi-level structural semantic knowledge. To preserve edge relation information and cohesive information in the graphs, we propose a topology-aware encoder for cohesive encoding. By learning from correlations and distinctions between texts and graphs via contrastive learning, SKIE can comprehend semantic associations and intrinsic patterns within the texts, enabling better adaptation to IE tasks.

Our contributions are as follows:

- We propose a novel pre-training method, which incorporates structural semantic knowledge from AMR graphs into the training process to enhance the capability and versatility of the resulting models, without additional annotation needs.
- To provide diverse structural knowledge, we elaborately design a topology-aware encoder, and then employ it to encode high-quality AMR cohesive subgraphs extracted according to two topology enhancement strategies.
- Experimental results demonstrate that our pre-training method achieves superior performance across multiple IE tasks, showcasing exceptional capabilities in both few-shot and zero-shot settings.

2 Related Work

2.1 Information Extraction

IE can be formulated as a text-to-structure task, with different IE subtasks corresponding to different target structures. OneIE (Lin et al., 2020) extracts optimal global information from input texts through global graph searching. Additionally, TANL (Paolini et al., 2021) translates structured prediction language tasks into IE processes through enhanced translation tasks between natural languages.

Recently, researchers have delved into universal frameworks for IE tasks. UIE (Lu et al., 2022) achieves generic modeling and adaptive structure generation for various IE tasks through structured language extraction and pattern-based prompting mechanisms. USM (Lou et al., 2023) decouples IE tasks and employs a unified semantic matching framework alongside unified token linking operations. UniEX (Ping et al., 2023) and UTC-IE (Yan et al., 2023) transform text-based IE tasks into a unified token-pair problem. UniEX leverages pattern-based cues and text information encoding, whereas UTC-IE achieves unified IE through axis-aware interactions and local interactions on the token-pair feature matrix.

2.2 Abstract Meaning Representation

AMR graph is a single-rooted directed graph used to represent the meaning of texts. AMR parsing translates texts into corresponding AMR graphs (Cai and Lam, 2020; Hoang et al., 2021; Wang et al., 2022; Vasylenko et al., 2023). With the continuous development of deep learning, there has been a gradual emergence of neural transition-based parsers, sequence-to-graph parsers, and sequence-to-sequence parsers (Bai et al., 2022).

The neural transition-based parsers (Fernandez Astudillo et al., 2020; Zhou et al., 2021; Drozdov et al., 2022) incrementally construct AMR graphs by applying basic operations (e.g., SHIFT,

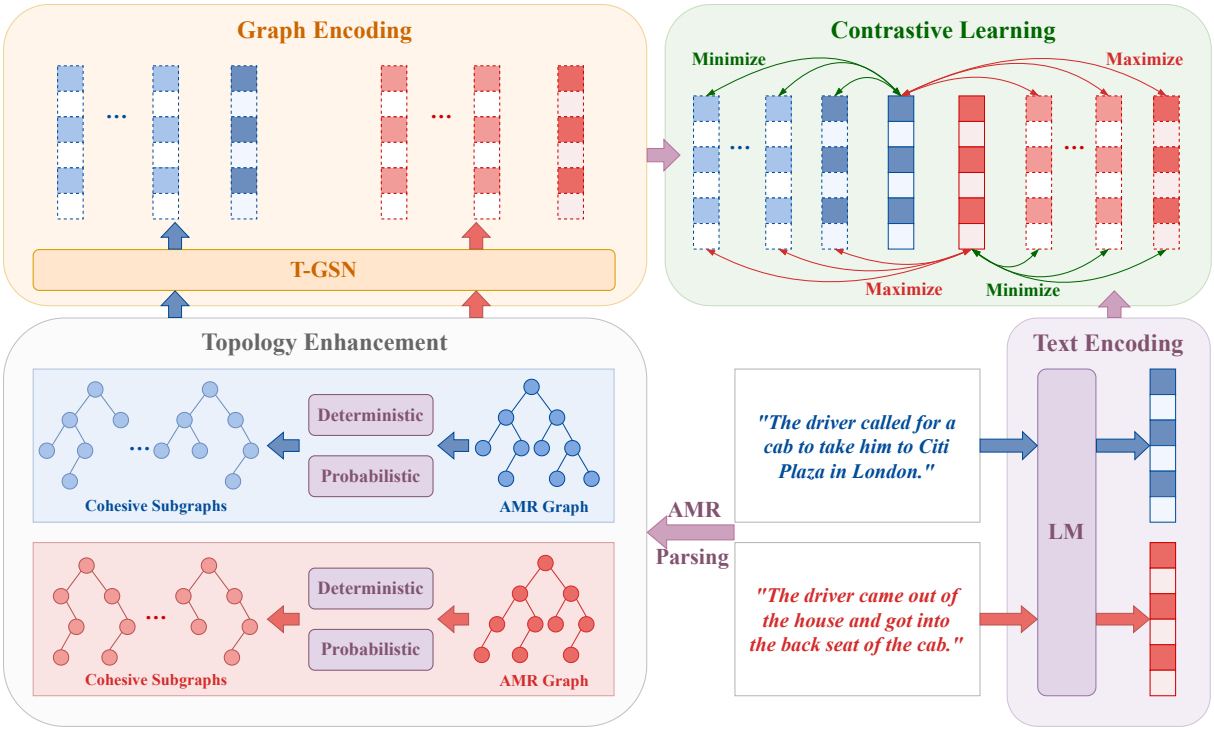

Figure 2: The overall framework of SKIE, which comprises three key modules: *topology enhancement*, *encoding cohesion*, and *contrastive learning*. The *topology enhancement* constructs cohesive subgraphs using both deterministic and probabilistic topology enhancement strategies. The *encoding cohesion* independently extracts features from texts and graphs. Finally, the *contrastive learning* analyzes the semantic correspondences between texts and graphs.

LEFT-ARC, RIGHT-ARC) in transition-based parsing. Sequence-to-graph parsers (Zhang et al., 2019; Cai and Lam, 2020; Xia et al., 2021) directly generate AMR graphs from texts. In addition, sequence-to-sequence parsers (Bevilacqua et al., 2021; Yu and Gildea, 2022; Gao et al., 2023) are employed to transform AMR parsing into a "linearized" sequence generation task.

3 Methodology

The overall framework is shown in Figure 2, consisting of three key modules: *topology enhancement*, *encoding cohesion*, and *contrastive learning*, which are introduced in section 3.1, section 3.2, and section 3.3, respectively. Section 3.4 describes how to fine-tune our PLM to adapt to IE tasks.

3.1 Topology Enhancement Module

3.1.1 AMR Parsing

For a given text s, we employ a transformer-based automatic AMR parser (Fernandez Astudillo et al., 2020) to obtain the corresponding AMR graph $G = (V, E)$. Here, nodes V represent basic semantic units such as entities and predicates, while edges E denote semantic relations. Each edge $e_{ij} = (v_i, v_j)$ is associated with a relation r from a predefined set of relations R, which can be formulated as $E = \{(v_i, v_j, r) \mid (v_i, v_j) \in V \times V, r \in R\}$.

3.1.2 Cohesive Subgraphs

After obtaining AMR graphs, we introduce cohesive subgraphs to derive semantic representations at different levels. These cohesive subgraphs aim to capture tight structures and semantic correlations within AMR graphs, revealing multi-level structural cohesion contained in the texts. We primarily focus on the *k-core* (Kong et al., 2019) due to its effectiveness in identifying core structures within graphs and its applicability to large-scale networks (King et al., 2023). For an AMR graph $G = (V, E)$, we extract a set of *k-core* cohesive subgraphs, denoted as $\mathbb{G} = \{G^k | k = k_{min}, k_{min+1}, \ldots, k_{max}\}$.

First, we utilize a **deterministic topological enhancement** strategy, which employs deterministic rules to generate subgraphs and select them based on predefined conditions. To further improve cohesion during graph diffusion, we strategically assign higher weights to edges within these cohesive subgraphs, thereby emphasizing key relations and enhancing overall structure connectivity.

For a node $v_i \in V$ in the original AMR graph, we obtain its importance weight $w_v(v_i)$ by calculating the number of times it appears in the subgraph set. Given $G^k = (V^k, E^k)$, we have:

$$w_v(v_i) = \sum_{G^k \in \mathbb{G}} \mathbf{1}_{v_i \in V^k} \quad (1)$$

where $\mathbf{1}_{v_i \in V^k}$ is an indicator function that outputs 1 if v_i is in V^k and 0 otherwise.

Then, the weight $w_v(v_i)$ is normalized to obtain $w_v^{'}(v_i)$, so that the uniformly initialized weight $w_e(e_{ij})$ of the edge e_{ij} between v_i and v_j can be continuously updated to derive weights $w_e^{'}(e_{ij})$ during graph diffusion:

$$w_e^{'}(e_{ij}) = \frac{1}{2}(w_v^{'}(v_i) + w_v^{'}(v_j))w_e(e_{ij}) \quad (2)$$

In this way, we refer to the deterministic topology enhancement strategies (Klicpera et al., 2019; Hassani and Khasahmadi, 2020; Wu et al., 2024) for graph diffusion:

$$\mathbf{S}^{PPR} = \alpha(\mathbf{I} - (1-\alpha)\mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2})^{-1} \quad (3)$$

where $\alpha \in (0, 1)$ is the teleport probability, $\mathbf{I}$ is the degree matrix of nodes, $\mathbf{D}$ is the diagonal degree matrix, and $\mathbf{A} \supseteq \mathbf{A}_{i,j}$ is the adjacency matrix with $\mathbf{A}_{i,j} = w_e^{'}(e_{ij})$.

Second, to compensate for the potential knowledge missing in deterministic topological enhancement, we incorporate a **probabilistic topological enhancement** strategy. This method typically generates subgraphs based on certain probability distributions or random processes, introducing randomness in subgraph generation, resulting in potentially different subgraphs each time.

Specifically, we introduce probabilistic mechanisms with the original dropping probability P to modify edges or nodes in the graph, thereby altering its topological structure. To maintain cohesion in the subgraphs generated by probabilistic topological enhancement, we employ a decay factor $\varepsilon \in (0, 1)$ to limit the probability P, which is initialized based on the node weights. The probabilities $P^{'}(v_i)$ for nodes v_i and $P^{'}(e_{ij})$ for edges e_{ij} are:

$$P^{'}(v_i) = (1 - w_v^{'}(v_i) \cdot \varepsilon) \cdot P \quad (4)$$

$$P^{'}(e_{ij}) = \frac{1}{2}(P^{'}(v_i) + P^{'}(v_j)) \quad (5)$$

Ultimately, we acquire a set of cohesive subgraphs that not only depict the multi-level structural cohesion of AMR graphs but also facilitate the capture of profound semantic knowledge from the original texts.

3.2 Encoding Cohesion Module

3.2.1 Text Encoder

To ensure a direct correspondence between AMR graphs and the original texts, we utilize Roberta-Large (Liu et al., 2019) as the text encoder, aligning its encoding structure with that of the AMR parser. Roberta-Large is a PLM based on Transformer architecture, known for its ability to effectively model diverse information across extensive unsupervised corpora. Given a text s, it is encoded through multiple layers of self-attention mechanisms to obtain the resulting vector from the last hidden layer. The vector encapsulate not only superficial information (such as vocabularies and phrases) but also contextual nuances.

3.2.2 Graph Encoder

We employ a graph encoder to extract corresponding vectors from AMR graphs and their cohesive subgraphs. However, traditional graph encoders exhibit shortcomings in handling such graphs in two primary aspects. First, they operate within the message-passing neural network framework, which concentrates on integrating neighbor information into a comprehensive representation, leading to the loss of substructure details. Second, since cohesive subgraphs are extracted based on node connectivity and feature-dense relations among nodes, it is crucial to preserve such semantic knowledge during the encoding process. Therefore, the graph encoder must prioritize maintaining both detailed substructure information and dynamic node relations.

In light of these limitations, we propose a **T**opology-aware **G**raph **S**ubstructure **N**etwork (T-GSN) that incorporates structural topology into GSN (Bouritsas et al., 2023). Specifically, we introduce relation-specific transformations to handle information uniquely according to the type of relations, allowing for tailored information processing from neighbors to derive the aggregated feature for each node. The update equation is articulated as follows:

$$\boldsymbol{h}_i^{(l+1)} = \sigma(\sum_{r \in R} \sum_{j \in \mathcal{N}_i^r} \frac{1}{n_{i,r}} \mathbf{W}_r^{(l)} \boldsymbol{h}_j^{(l)} + \mathbf{W}_o^{(l)} \boldsymbol{h}_i^{(l)}) \quad (6)$$

where $\boldsymbol{h}_i^{(l)}$ denotes the feature of node v_i at layer l. σ represents the activation function, such as ReLU. $\mathcal{N}_i^r$ represents the set of neighbor nodes of node

v_i under relation r. $\mathbf{W}_r^{(l)}$ denotes the weight matrix for relation r at layer l, and $\mathbf{W}_o^{(l)}$ represents the self-connection weight matrix to maintain own features of the nodes. $n_{i,r}$ is the normalization constant, typically chosen as $|\mathcal{N}_i^r|$, which represents the number of neighbors of node v_i associated with relation r.

Then, the updated feature $\boldsymbol{h}_i^{(l+1)}$ and the encoded feature x_i of node v_i in the AMR graph are fed into T-GSN:

$$\text{T-GSN}(v_i) = \text{AGG}(\boldsymbol{h}_i^{(l+1)}, \boldsymbol{h}_j^{(l+1)}, \boldsymbol{x}_i, \boldsymbol{x}_j) \quad (7)$$

where AGG denotes a neighborhood aggregation function, which may involve utilizing a multi-layer perceptron to aggregate features of node v_i from its neighbors $j \in \mathcal{N}_i^r$.

Ultimately, we encode the graphs not only based on their topological structures but also by considering the semantic relations within substructures. T-GSN significantly amplifies the expressive capabilities of GNNs, allowing them to more accurately capture and comprehend the intricate structures and representations of AMR graphs and their cohesive subgraphs.

3.3 Contrastive Learning Module

Contrastive learning has made significant strides in the field of representation learning, particularly demonstrating outstanding results in self-supervised training. It learns the intrinsic representation of data by ***maximizing*** the distance among negative sample pairs and ***minimizing*** the distance among positive sample pairs. During pre-training, contrastive learning can help models distinguish between graph-text pairs with different similarities, enabling them to more accurately capture the correspondence between texts and graphs.

Specifically, we employ triplet loss (Schroff et al., 2015) as our contrastive learning loss function, structured in triplets <*anchor, positive, negative*>. For a given text s, we designate it as an anchor, forming positive pairs with its corresponding AMR graph and AMR cohesive subgraphs, while generating negative pairs with non-matching AMR graphs and AMR cohesive subgraphs. Our core concept revolves around ensuring a certain margin separation between positive and negative pairs. This optimization ensures that samples of the same category in the embedding space are sufficiently close, while samples of different categories are adequately distant. In essence, the distance between the anchor sample and the negative sample should significantly exceed the distance between the anchor and the positive sample:

$$L = \max(0, |\boldsymbol{s} - \boldsymbol{g}_+|^2 - |\boldsymbol{s} - \boldsymbol{g}_-|^2 + \text{m}) \quad (8)$$

here, $\boldsymbol{s}$, $\boldsymbol{g}_+$, and $\boldsymbol{g}_-$ respectively represent the vectors that map the text s, the positive graphs corresponding to the text, and the negative graphs into the embedding space. $|\cdot|$ denotes the Euclidean distance (or other distance metric), and m is a positive number defining the minimum separation between positive and negative sample pairs.

SKIE can better comprehend the semantic correspondence between texts and graphs by minimizing loss to enhance performance in downstream IE tasks. This effect is particularly pronounced when dealing with semantically complex texts.

3.4 Task-specific Fine-tuning

Since the focus of this paper is on pre-training rather than fine-tuning, we adopt fine-tuning techniques from previous work (Zhu et al., 2023) to efficiently adapt the PLM to different IE tasks and settings. First, we convert the text s into an input t_i in a unified data format by embedding an instruction and a schema label. The instruction starts with a leading token `[I]` and includes a sentence to prompt the model with a specific task (e.g., *"Please extract event information from the given text, including triggers and arguments"*). The schema label serves as guidance for different IE tasks (e.g., `[LM]` for entities or event types and `[LR]` for relations or argument roles).

Then, we use the PLM to convert t_i into the vector $\boldsymbol{z}_i \in \mathbb{R}^{d_z}$ and obtain the adjacency matrix $\mathbf{B}$ of the multi-span cyclic graph through biaffine attention (Dozat and Manning, 2017). The multi-span cyclic graph includes three types of connections: consecutive connections within the same entity span, jump connections linking different slots within tuples, and tail-to-head connections marking the boundaries of the graph. The connection probability p_{ij}^c ($c \in$ {continuous, jump, tail-to-head}) between t_i and t_j in the matrix $\mathbf{B}$ ($p_{ij}^c > 0.5, \mathbf{B}_{ij}^c = 1$, otherwise $\mathbf{B}_{ij}^c = 0$) is formulated as:

$$p_{ij}^c = \text{sigmoid}(\boldsymbol{z}_i'^{\top} U \boldsymbol{z}_j' / \sqrt{d_z}) \quad (9)$$

where $\boldsymbol{z}_i' = \text{FFNN}_s(\boldsymbol{z}_i)$, $\boldsymbol{z}_j' = \text{FFNN}_e(\boldsymbol{z}_j)$, $U \in \mathbb{R}^{d_b \times 3 \times d_b}$ is the trainable parameter, with d_b denoting the biaffine size, 3 representing three types of

Task	Datasets	TANL	UIE	UniEX	USM	UTC-UIE	Mirror	MetaRetriever	SKIE
NER	ACE04	-	86.89	87.12	87.62	87.54	87.16	86.10	**88.12**
	ACE05	84.90	85.78	87.02	87.14	87.75	85.34	84.01	**88.52**
	CoNLL03	91.70	92.99	92.65	93.16	93.45	92.73	92.38	**93.62**
RE	ACE05	63.70	66.06	66.06	67.88	67.79	67.86	63.37	**72.36**
	CoNLL04	71.40	75.00	73.40	78.84	-	75.22	73.66	**78.91**
	SciERC	-	36.53	38.00	37.36	38.77	36.89	35.77	**46.90**
EE	ACE05-Tgg	68.40	73.36	74.08	72.41	73.46	74.44	72.38	**75.15**
	ACE05-Arg	47.60	54.79	53.92	55.83	56.51	55.88	52.62	**61.77**
	CASIE-Tgg	-	69.33	71.46	71.73	-	71.81	69.76	**71.95**
	CASIE-Arg	-	61.30	62.91	63.26	-	61.27	60.37	**63.96**

Table 1: Overall F1-scores on 8 IE benchmarks (-Tgg. and -Arg. denote event trigger and arguments, respectively). These datasets are excluded from the pre-training phase. The results of UIE are reported based on the UIE-Large model proposed by Lu et al. (2022). The best results are shown in bold.

connections. FFNN is a feedforward neural network incorporating rotary positional embeddings, as introduced in RoFormer (Su et al., 2024).

Finally, we use Circle Loss (Su et al., 2022) as the downstream loss function:

$$\mathcal{L}(i,j) = \log\left(1 + \sum_{\text{neg}} e^{p^c_{ij}}\right) + \log\left(1 + \sum_{\text{pos}} e^{-p^c_{ij}}\right) \tag{10}$$

where neg stands for negative samples and pos denotes positive samples.

4 Experiments

4.1 Experiment Setup

4.1.1 Datasets

We collect a large amount of datasets and transform them into unified unsupervised corpora for pre-training. A detailed list of the pre-training corpora can be found in Appendix A.

We conduct experiments on NER, RE, and EE tasks, including 8 IE benchmarks: ACE04 (Mitchell et al., 2005), ACE05 (Walker et al., 2006), CoNLL03 (Tjong Kim Sang and De Meulder, 2003), CoNLL04 (Roth and Yih, 2004), SciERC (Luan et al., 2018), and CASIE (Satyapanich et al., 2020). Additionally, we employ five subsets of CrossNER (AI, literature, music, politics, and science) (Liu et al., 2021) to evaluate the zero-shot capabilities of SKIE. All extraction tasks adopt an end-to-end setting, taking texts as input and directly generating the target structure.

4.1.2 Baselines

We compare SKIE with generation-based TANL (Paolini et al., 2021), UIE (Lu et al., 2022), MetaRetriever (Cong et al., 2023), and extraction-based UniEX (Ping et al., 2023), USM (Lou et al., 2023), UTC-IE (Yan et al., 2023), Mirror (Zhu et al., 2023), respectively. Among them, UIE, USM, Mirror, and MetaRetriever are all pre-training methods for IE tasks.

During pre-training, we tune the graph encoding layers, the margin in the triplet loss, and the decay factor ε in the topology enhancement strategy to improve training outcomes. The implementation details of the pre-training and fine-tuning phases are included in Appendix B. The code is available at `https://anonymous.4open.science/r/SKIE`.

4.1.3 Evaluation

We employ span-based offset Micro-F1 as the primary metric to evaluate methods for different IE tasks: For NER tasks, an entity is considered correct if its offset and type are correct. For RE tasks, under strict matching, a relation is correct if the relation type, the offsets, and the types of related entities are correct. For event trigger extraction tasks, an event trigger is considered correct if its offset and event type match the reference trigger. For event argument extraction tasks, an event argument is correct if its offset, role type, and event type match the reference argument.

Task	Few-Shot	UIE	USM	Mirror	MetaRetriever	SKIE
NER (CoNLL03)	1-shot	57.53	71.11	76.49	49.44	**77.50**
	5-shot	75.32	83.25	82.45	69.88	**83.75**
	10-shot	79.12	84.58	84.69	74.19	**85.46**
	Avg.	70.66	79.65	81.21	64.50	**82.24**
RE (CoNLL04)	1-shot	34.88	36.17	26.29	29.90	**37.54**
	5-shot	51.64	53.20	47.42	47.02	**55.70**
	10-shot	58.98	60.99	55.77	53.95	**61.31**
	Avg.	48.50	50.12	43.16	43.62	**51.52**
Event Trigger (ACE05-Evt)	1-shot	42.37	40.86	47.77	39.85	**48.19**
	5-shot	53.07	55.61	57.90	49.43	**58.21**
	10-shot	54.35	58.79	59.16	53.58	**66.27**
	Avg.	49.93	51.75	54.94	47.62	**57.56**
Event Argument (ACE05-Evt)	1-shot	14.56	19.01	23.18	13.30	**23.76**
	5-shot	31.20	36.69	**37.74**	27.70	35.13
	10-shot	35.19	42.48	39.20	32.31	**43.84**
	Avg.	26.98	32.73	33.38	24.44	**34.24**

Table 2: Few-shot results on IE tasks. Avg. denotes the average performance over 1/5/10-shot. The results of UIE are reported based on the UIE-Large model proposed by Lu et al. (2022). The best results are shown in bold.

4.2 Main Results

Table 1 shows the performance of all methods on the aforementioned IE benchmarks. Compared to other baselines, SKIE outperforms them across all datasets, achieving an average F1-score improvement of 1.49, 6.75, and 3.42 in NER, RE, and EE tasks, respectively. This strongly demonstrates the effectiveness of our pre-training method, which leverages structural semantic knowledge to enhance the performance on IE tasks.

Meanwhile, our pre-training corpora supplement more RE and EE datasets compared to pre-training methods such as Mirror (Zhu et al., 2023), resulting in a significant improvement in downstream RE and EE tasks. SKIE can rapidly adapt to downstream IE tasks, enabling efficient and targeted extractions. Notably, our pre-training corpora do not require manually setting prompts or annotations, substantially reducing labor costs and facilitating the integration of new corpora in the future. Additionally, it avoids the impact of label errors or label drift on training.

4.3 Few-shot Results

We focus on the performance of SKIE in low-resource settings. To validate its rapid adaptation capability, we conduct few-shot experiments. Specifically, we sample 1/5/10 texts per entity/relation/event type in the training set, following the experimental setup of previous work (Lou et al., 2023). To mitigate the impact of random sampling, each experiment is repeated 10 times with different samples, and the average F1-score is used to represent performances.

As shown in Table 2, SKIE achieves excellent results on CoNLL03, CoNLL04, and ACE05. Among the four tasks, the NER task is relatively easier to handle and can achieve satisfactory results with minimal fine-tuning. However, for tasks associated with other datasets, there is a significant gap between the few-shot fine-tuning results and the full fine-tuning results, highlighting the difficulty of these tasks and the effectiveness of fine-tuning. Additionally, SKIE can learn deeper structural semantic knowledge during pre-training, rather than capturing information specific to a particular task. Therefore, compared to baselines with limited samples, SKIE performs better on these tasks even with only a few samples.

4.4 Zero-shot Results

Table 3 shows the zero-shot results of SKIE on 5 NER datasets, which are eliminated during pre-training. SKIE outperforms USM and Mirror on

most of datasets, achieving a superior average F1-score of 58.03, notably exceeding USM. Emphasized that USM trains on the same datasets and evaluates using the provided labels, while SKIE is not exposed to these datasets before testing.

Among the above datasets, SKIE achieves the most outstanding performance enhancement on literature, with an average F1-score improvement of 10.43, due to using a dataset containing academic content during pre-training. However, SKIE falls short of Mirror in the politics dataset, with a lower F1-score of 3.77, suggesting that enhancing the pre-training with more diverse and comprehensive data could potentially improve SKIE's performance.

Datasets	USM	Mirror	SKIE
AI	28.18	45.23	**52.45**
Literature	56.00	46.32	**56.75**
Music	44.93	58.61	**59.87**
Politics	36.10	**67.30**	63.53
Science	44.09	54.84	**57.57**
Avg.	41.98	54.46	**58.03**

Table 3: Zero-shot results on 5 NER datasets, which are eliminated during pre-training. The best results are shown in bold.

4.5 Ablation Results

To validate the effectiveness of SKIE, we explore the impact of modifications to the graph encoder and topological enhancement strategies. As shown in Table 4, GSN can capture local substructure features in graphs more precisely, rather than GCN focusing solely on global features. However, GSN performs worse than T-GSN, resulting in an average F1-score drop of 7.69, proving that modifying the graph encoder enables better capture of the cohesive information in AMR graphs and preserves edge relation information.

Additionally, the results reveal that a single deterministic topological enhancement may lead to knowledge missing, while a single probabilistic topological enhancement may shift the cohesive center, thereby affecting the quality of the generated cohesive subgraphs. Meanwhile, the average F1-score without cohesive subgraphs decreases by 6.97. This indicates that cohesive subgraphs introduce multi-level structural semantic knowledge during pre-training, markedly enhancing the effectiveness and generalization versatility of SKIE. When removing both cohesive subgraphs and T-GSN, there are expressive declines in performance across IE tasks, underscoring the essential roles of these components.

Model	NER	RE	EE Tgg.	EE Arg.
w/ GSN	75.71	70.43	69.32	51.77
w/ GCN	64.73	47.75	57.10	27.16
w/o PTE	81.18	66.41	71.05	59.87
w/o DTE	87.18	67.81	72.42	58.99
w/o CS	76.47	65.85	70.01	57.61
w/o All	70.28	62.74	54.32	45.60
SKIE	88.52	72.36	75.15	61.77

Table 4: Ablation results of SKIE on ACE05. "w/ GSN" and "w/ GCN" refer to pre-training with a standard GSN/GCN encoder. "w/o PTE" and "w/o DTE" indicate the exclusion of probabilistic/deterministic topological enhancement. "w/o CS" denotes pre-training removal of cohesive subgraphs. "w/o all" describes pre-training without both subgraphs and the T-GSN encoder.

4.6 Corpora Validity Results

To evaluate the quality of the corpora used for pre-training, we conduct an ablation study on different types of pre-training data, as shown in Table 5. The results indicate that removing any part of pre-training data negatively impacts the performance, demonstrating the effectiveness of our pre-training corpora. Additionally, it can be seen that pre-training with a combination of different types of pre-training data, rather than relying solely on a single type, improves the performance of the corresponding downstream tasks. This suggests that there is indeed a correlation between different IE tasks. Therefore, it is beneficial to perform joint IE tasks, as this facilitates mutual learning among different IE tasks, leading to better extraction results.

Corpus	NER	RE	EE Tgg.	EE Arg.
only NER	84.86	57.80	64.98	53.31
only RE	84.03	67.73	68.27	50.37
only EE	86.47	60.72	69.44	52.70
All	88.52	72.36	75.15	61.77

Table 5: Pre-training corpora validity results on ACE05, which exclusively include NER, RE, or EE datasets.

4.7 Language Adaptation Results

To verify language adaptability, we evaluate SKIE on Multiconel (Malmasi et al., 2022), which is a common multilingual dataset for NER. Table 6 shows the results of our English AMR parser based model compared to ChatGPT and GLiNER (Zaratiana et al., 2024). GLiNER-En and GLiNER-Multi are two variants of GLiNER, utilizing two versions of deBERTa-v3: GLiNER-En uses deBERTa-v3-Large, while GLiNER-Multi employs mdeBERTa-v3-base, which is the multilingual version of deBERTa-v3. It can be seen that even using an English AMR parser and pre-training on English corpora, our model can still achieve satisfactory IE performance on other languages, demonstrating the generalizability of SKIE.

Language	ChatGPT	GLi-En	GLi-Multi	SKIE
German	37.1	35.6	39.5	**67.5**
English	37.2	42.4	41.7	**71.9**
Spanish	34.7	38.7	42.1	**58.5**
Dutch	35.7	35.6	38.9	**41.4**
Bengali	23.3	0.89	25.9	**34.1**
Persian	25.9	14.9	30.2	**31.6**
Hindi	27.3	11.3	27.8	**29.4**
Korean	30.0	20.5	**28.7**	28.6
Russian	27.4	30.3	33.3	**37.5**
Turkish	31.9	22.0	30.0	**33.6**
Chinese	18.8	6.59	24.3	**25.8**

Table 6: Language adaptation results on Multiconel. The results of ChatGPT are taken from Lai et al. (2023). GLi-En represents GLiNER-En which employs deBERTa-v3-Large, and GLi-Multi represents GLiNER-Multi using mdeBERTa-v3-base.

5 Conclusion

In this paper, we propose SKIE, a contrastive pre-training method designed to enhance IE models with structural semantic knowledge. Specifically, SKIE leverages AMR graphs generated from unsupervised texts as self-supervised signals and further extracts cohesive subgraphs to provide multi-level structural semantic knowledge. Additionally, SKIE integrates edge relation information and cohesion information for the encoder, effectively enhancing the learning process of PLMs. Compared to existing methods, SKIE enables the training on unsupervised datasets in a self-supervised manner, significantly reducing the annotation burden. The resulting models demonstrate proficiency in handling IE tasks on complex texts by utilizing the structural semantic knowledge. Experimental results show that SKIE achieves state-of-the-art performances across multiple IE tasks and excels in few-shot and zero-shot settings. Our future work will focus on refining SKIE to alleviate noise in AMR graphs and extending its application to broader NLP tasks.

Limitations

Although SKIE has shown outstanding performance in IE tasks, it still has some limitations. Firstly, we use nearly a million datasets during pre-training, and the need to encode both texts and graphs separately resulted in lengthy runtime. Secondly, due to the constraints of existing public datasets, the NER, RE, and EE pre-training datasets we found are imbalanced, with the EE dataset being much smaller in scale compared to NER and RE, limiting the performance of the EE task. Finally, the Roberta-Large model we used has a maximum input sequence length of 512 tokens. However, IE tasks often require processing longer texts. In the future, we will consider using a sliding window approach to handle the input or exploring other models capable of processing longer sequences, such as Longformer or BigBird.

Ethics Statement

In the development of our pre-training framework, we acknowledge several ethical considerations. Our method requires large-scale corpora collected from the Internet, which may exhibit common domain biases (e.g., in the news domain). Such biases can lead to errors in domain-specific IE tasks, potentially causing inaccuracies in real-world applications and affecting the reliability of the resulting model. It is crucial to recognize and address these potential ethical issues to ensure that our method is used responsibly and ethically in real-world applications.

Acknowledgments

The authors would like to thank anonymous reviewers for their helpful comments and suggestions. This work is supported by the National Key R&D Program of China (2022YFB3103202).

References

Xuefeng Bai, Yulong Chen, and Yue Zhang. 2022. Graph pre-training for AMR parsing and generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6001–6015, Dublin, Ireland. Association for Computational Linguistics.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract Meaning Representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186, Sofia, Bulgaria. Association for Computational Linguistics.

Michele Bevilacqua, Rexhina Blloshmi, and Roberto Navigli. 2021. One spring to rule them both: Symmetric amr semantic parsing and generation without a complex pipeline. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(14):12564–12573.

Giorgos Bouritsas, Fabrizio Frasca, Stefanos Zafeiriou, and Michael M. Bronstein. 2023. Improving graph neural network expressivity via subgraph isomorphism counting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1):657–668.

Deng Cai and Wai Lam. 2020. AMR parsing via graph-sequence iterative inference. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1290–1301, Online. Association for Computational Linguistics.

Xin Cong, Bowen Yu, Mengcheng Fang, Tingwen Liu, Haiyang Yu, Zhongkai Hu, Fei Huang, Yongbin Li, and Bin Wang. 2023. Universal information extraction with meta-pretrained self-retrieval. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4084–4100, Toronto, Canada. Association for Computational Linguistics.

Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. In *International Conference on Learning Representations*.

Andrew Drozdov, Jiawei Zhou, Radu Florian, Andrew McCallum, Tahira Naseem, Yoon Kim, and Ramón Astudillo. 2022. Inducing and using alignments for transition-based AMR parsing. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1086–1098, Seattle, United States. Association for Computational Linguistics.

Ramón Fernandez Astudillo, Miguel Ballesteros, Tahira Naseem, Austin Blodgett, and Radu Florian. 2020. Transition-based parsing with stack-transformers. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1001–1007, Online. Association for Computational Linguistics.

Bofei Gao, Liang Chen, Peiyi Wang, Zhifang Sui, and Baobao Chang. 2023. Guiding AMR parsing with reverse graph linearization. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13–26, Singapore. Association for Computational Linguistics.

Sreyan Ghosh, Utkarsh Tyagi, Manan Suri, Sonal Kumar, Ramaneswaran S, and Dinesh Manocha. 2023. ACLM: A selective-denoising based generative data augmentation approach for low-resource complex NER. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 104–125, Toronto, Canada. Association for Computational Linguistics.

Ralph Grishman. 2019. Twenty-five years of information extraction. *Natural Language Engineering*, 25(6):677–692.

Ralph Grishman and Beth Sundheim. 1996. Message Understanding Conference- 6: A brief history. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*.

Luis Guzman Nateras, Franck Dernoncourt, and Thien Nguyen. 2023. Hybrid knowledge transfer for improved cross-lingual event detection via hierarchical sample selection. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5414–5427, Toronto, Canada. Association for Computational Linguistics.

Kaveh Hassani and Amir Hosein Khasahmadi. 2020. Contrastive multi-view representation learning on graphs. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org.

Thanh Lam Hoang, Gabriele Picco, Yufang Hou, Young-Suk Lee, Lam Nguyen, Dzung Phan, Vanessa Lopez, and Ramon Fernandez Astudillo. 2021. Ensembling graph predictions for amr parsing. In *Advances in Neural Information Processing Systems*, volume 34, pages 8495–8505. Curran Associates, Inc.

Valerie King, Alex Thomo, and Quinton Yong. 2023. Computing (1+epsilon)-approximate degeneracy in sublinear time. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*, pages 2160–2168. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Johannes Klicpera, Stefan Weißenberger, and Stephan Günnemann. 2019. Diffusion improves graph learning. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 13333–13345.

Yi-Xiu Kong, Gui-Yuan Shi, Rui-Jie Wu, and Yi-Cheng Zhang. 2019. k-core: Theories and applications. *Physics Reports*, 832:1–32. K-core: Theories and Applications.

Viet Dac Lai, Nghia Trung Ngo, Amir Pouran Ben Veyseh, Hieu Man, Franck Dernoncourt, Trung Bui, and Thien Huu Nguyen. 2023. Chatgpt beyond english: Towards a comprehensive evaluation of large language models in multilingual learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 13171–13189. Association for Computational Linguistics.

Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A joint neural model for information extraction with global features. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7999–8009, Online. Association for Computational Linguistics.

Jian Liu, Dianbo Sui, Kang Liu, Haoyan Liu, and Zhe Zhao. 2023. Learning with partial annotations for event detection. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 508–523, Toronto, Canada. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Zihan Liu, Yan Xu, Tiezheng Yu, Wenliang Dai, Ziwei Ji, Samuel Cahyawijaya, Andrea Madotto, and Pascale Fung. 2021. Crossner: Evaluating cross-domain named entity recognition. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 13452–13460. AAAI Press.

Jie Lou, Yaojie Lu, Dai Dai, Wei Jia, Hongyu Lin, Xianpei Han, Le Sun, and Hua Wu. 2023. Universal information extraction as unified semantic matching. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'23/IAAI'23/EAAI'23. AAAI Press.

Yaojie Lu, Qing Liu, Dai Dai, Xinyan Xiao, Hongyu Lin, Xianpei Han, Le Sun, and Hua Wu. 2022. Unified structure generation for universal information extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5755–5772, Dublin, Ireland. Association for Computational Linguistics.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3219–3232, Brussels, Belgium. Association for Computational Linguistics.

Shervin Malmasi, Anjie Fang, Besnik Fetahu, Sudipta Kar, and Oleg Rokhlenko. 2022. Multiconer: A large-scale multilingual dataset for complex named entity recognition. In *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022, Gyeongju, Republic of Korea, October 12-17, 2022*, pages 3798–3809. International Committee on Computational Linguistics.

Alexis Mitchell, Stephanie Strassel, Shudong Huang, and Ramez Zakhary. 2005. *ACE 2004 Multilingual Training Corpus*.

Giovanni Paolini, Ben Athiwaratkun, Jason Krone, Jie Ma, Alessandro Achille, RISHITA ANUBHAI, Cicero Nogueira dos Santos, Bing Xiang, and Stefano Soatto. 2021. Structured prediction as translation between augmented natural languages. In *International Conference on Learning Representations*.

Yang Ping, JunYu Lu, Ruyi Gan, Junjie Wang, Yuxiang Zhang, Pingjian Zhang, and Jiaxing Zhang. 2023. UniEX: An effective and efficient framework for unified information extraction via a span-extractive perspective. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16424–16440, Toronto, Canada. Association for Computational Linguistics.

Dan Roth and Wen-tau Yih. 2004. A linear programming formulation for global inference in natural language tasks. In *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004*, pages 1–8, Boston, Massachusetts, USA. Association for Computational Linguistics.

Oscar Sainz, Iker García-Ferrero, Rodrigo Agerri, Oier Lopez de Lacalle, German Rigau, and Eneko Agirre. 2024. Gollie: Annotation guidelines improve zero-shot information-extraction. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Taneeya Satyapanich, Francis Ferraro, and Tim Finin. 2020. Casie: Extracting cybersecurity event information from text. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8749–8757.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. 2023. DiffusionNER: Boundary diffusion for named entity recognition. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3875–3890, Toronto, Canada. Association for Computational Linguistics.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063.

Jianlin Su, Ahmed Murtadha, Shengfeng Pan, Jing Hou, Jun Sun, Wanwei Huang, Bo Wen, and Yunfeng Liu. 2022. Global pointer: Novel efficient span-based approach for named entity recognition. *Preprint*, arXiv:2208.03054.

Qi Sun, Kun Huang, Xiaocui Yang, Pengfei Hong, Kun Zhang, and Soujanya Poria. 2023. Uncertainty guided label denoising for document-level distant relation extraction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15960–15973, Toronto, Canada. Association for Computational Linguistics.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Pavlo Vasylenko, Pere Lluís Huguet Cabot, Abelardo Carlos Martínez Lorenzo, and Roberto Navigli. 2023. Incorporating graph information in transformer-based AMR parsing. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1995–2011, Toronto, Canada. Association for Computational Linguistics.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. *ACE 2005 Multilingual Training Corpus*.

Chuan Wang, Nianwen Xue, and Sameer Pradhan. 2015. Boosting transition-based AMR parsing with refined actions and auxiliary analyzers. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 857–862, Beijing, China. Association for Computational Linguistics.

Peiyi Wang, Liang Chen, Tianyu Liu, Damai Dai, Yunbo Cao, Baobao Chang, and Zhifang Sui. 2022. Hierarchical curriculum learning for AMR parsing. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 333–339, Dublin, Ireland. Association for Computational Linguistics.

Shengqiong Wu, Hao Fei, Yixin Cao, Lidong Bing, and Tat-Seng Chua. 2023. Information screening whilst exploiting! multimodal relation extraction with feature denoising and multimodal topic modeling. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14734–14751, Toronto, Canada. Association for Computational Linguistics.

Yucheng Wu, Leye Wang, Xiao Han, and Han-Jia Ye. 2024. Graph contrastive learning with cohesive subgraph awareness. In *Proceedings of the ACM on Web Conference 2024*, WWW '24, page 629–640, New York, NY, USA. Association for Computing Machinery.

Qingrong Xia, Zhenghua Li, Rui Wang, and Min Zhang. 2021. Stacked AMR parsing with silver data. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4729–4738, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Hang Yan, Yu Sun, Xiaonan Li, Yunhua Zhou, Xuanjing Huang, and Xipeng Qiu. 2023. UTC-IE: A unified token-pair classification architecture for information extraction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4096–4122, Toronto, Canada. Association for Computational Linguistics.

Chen Yu and Daniel Gildea. 2022. Sequence-to-sequence AMR parsing with ancestor information. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 571–577, Dublin, Ireland. Association for Computational Linguistics.

Urchade Zaratiana, Nadi Tomeh, Pierre Holat, and Thierry Charnois. 2024. Gliner: Generalist model for named entity recognition using bidirectional transformer. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, pages 5364–5376. Association for Computational Linguistics.

Sheng Zhang, Xutai Ma, Kevin Duh, and Benjamin Van Durme. 2019. AMR parsing as sequence-to-graph transduction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 80–94, Florence, Italy. Association for Computational Linguistics.

Jiawei Zhou, Tahira Naseem, Ramón Fernandez Astudillo, and Radu Florian. 2021. AMR parsing with action-pointer transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5585–5598, Online. Association for Computational Linguistics.

Wenxuan Zhou, Sheng Zhang, Yu Gu, Muhao Chen, and Hoifung Poon. 2024. Universalner: Targeted distillation from large language models for open named entity recognition. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Tong Zhu, Junfei Ren, Zijian Yu, Mengsong Wu, Guoliang Zhang, Xiaoye Qu, Wenliang Chen, Zhefeng Wang, Baoxing Huai, and Min Zhang. 2023. Mirror: A universal framework for various information extraction tasks. In *Proceedings of the 2023 Conference*

on Empirical Methods in Natural Language Processing, pages 8861–8876, Singapore. Association for Computational Linguistics.

A Pre-training Corpora Statistics

Tables 7, 8, and 9 present the detailed statistics of the pre-training corpora containing NER, RE, or EE datasets. The datasets underwent a comprehensive cleaning process to ensure data quality and relevance for training purposes, including removing irregular symbols, eliminating non-English sentences, and deleting excessively short sentences.

Name	Instance
AnatEM	5,442
bc2gm	12,088
bc4chemd	30,468
Broad Tweet	329
FabNER	6,595
FindVehicle	21,565
GENIA	8,717
GUM	9,493
HarveyNER	3,768
MIT-movie	8,881
MIT-restaurant	6,669
MultiCoNER	3,388
MultiNERD	80,592
OntoNotes5	49,442
SEC-filings	1,010
TweetNER7	126
WNUT-16	2,310
WNUT-17	3,258
Total	262,138

Table 7: Detailed statistics of NER datasets used for pre-training.

Additionally, we implement deduplication to eliminate repetitive data entries and conduct tokenization of the text for better processing and analysis. These steps are critical in preparing the datasets for effective pre-training, aligning with best practices in data preprocessing for machine learning applications.

Name	Instance
ADE	4,252
FewRel	44,733
GIDS	11,290
kbp37	15,908
NYT10	508,629
NYT10-HRL	70,225
NYT11-HRL	235,750
WebNLG	21,170
Wiki80	5,187
Total	917,144

Table 8: Detailed statistics of RE datasets used for pre-training.

Name	Instance
GENEVA	3,670
MLEE	2,618
PHEE	2,872
RAMS	50,331
WikiEvents	6,129
Total	65,620

Table 9: Detailed statistics of EE datasets used for pre-training.

B Implementation Details

We conduct experiments on the same NVIDIA Tesla A100 GPU. The hyper-parameter configurations for pre-training and fine-tuning are detailed in Table 10 and 11, respectively. Figures 3, 4, and 5 present the loss trends over 30 epochs during pre-training across different hyper-parameter settings.

Figure 3 examines the impact of decay factors set at 0.1, 0.2, and 0.3, showing that the decay factor of 0.2 leads to optimal loss reduction over time. Figure 4 illustrates the influence of varying the number of graph encoder layers (2, 3, and 4 layers). Here, the configuration with 3 layers demonstrates the most effective learning. Figure 5 explores the effects of different margin values (0.1, 0.2, and 0.3). The results indicate that the margin of 0.1 achieves the most consistent reduction in loss. These analyses confirm that the optimal

Parameters	Setting
decay factor ε	0.2
graph encoder Layers	3
k	5
margin	0.1
text encoder learning rate	1e-5
graph encoder learning rate	1e-3
epochs	50
batch size	64
d_h	128

Table 10: Pre-training hyper-parameter settings.

Parameters	Setting
warmup proportion	0.1
epochs	10
epoch patience	3
few-shot epochs	200
batch size	64
PLM learning rate	2e-5
other learning rate	1e-3
max gradient norm	0.5
d_h	1024
d_b	512
dropout	0.4

Table 11: Fine-tuning hyper-parameter settings.

hyper-parameter configurations for our pre-training process include the decay factor of 0.2, the graph encoder layers of 3, and the margin of 0.1.

C K-core

A k-core cohesive subgraph is a maximal subgraph in which every node is connected to at least k other nodes. The core idea of the k-core algorithm is to identify core nodes in the graph by iteratively removing nodes with degrees less than k and their associated edges. The steps are as follows:

Step 1-Initialization: Calculate the degree of each node in the graph and store the degrees in a dictionary.

Step 2-Iterative pruning: Remove all nodes with degrees less than k and their incident edges from the graph, resulting in a new subgraph. Then, repeat the same operation on this new subgraph until no more nodes can be removed.

Step 3-Result output: The final subgraph obtained is the k-core cohesive subgraph.

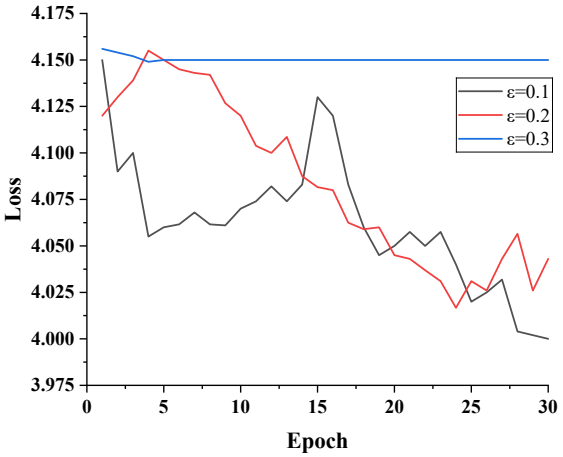

Figure 3: The loss trends during pre-training with different decay factor ε settings.

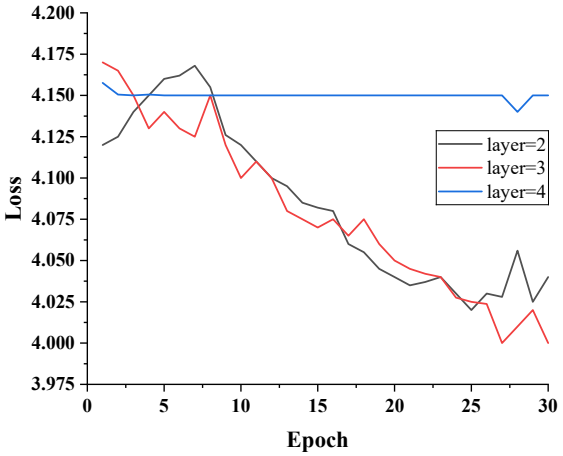

Figure 4: The loss trends during pre-training with different graph encoding layers settings.

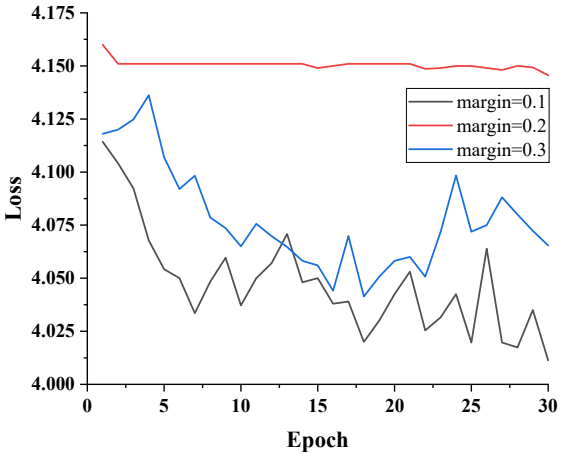

Figure 5: The loss trends during pre-training with different margin settings.

Datasets	UniNER (7B)	GoLLIE (7B)	GLiNER-L (0.3B)	Mirror-RL (0.3B)	SKIE (0.3B)
AI	53.60	**59.10**	57.20	48.95	52.45
Literature	59.30	62.70	**64.40**	50.11	56.75
Music	67.00	67.80	**69.60**	59.60	59.87
Politics	60.90	57.20	**72.60**	56.80	63.53
Science	61.10	55.50	**62.60**	55.29	57.57
Avg.	60.38	60.46	**65.28**	54.15	58.03

Table 12: Supplementary zero-shot results on 5 NER datasets. The best results are shown in bold.

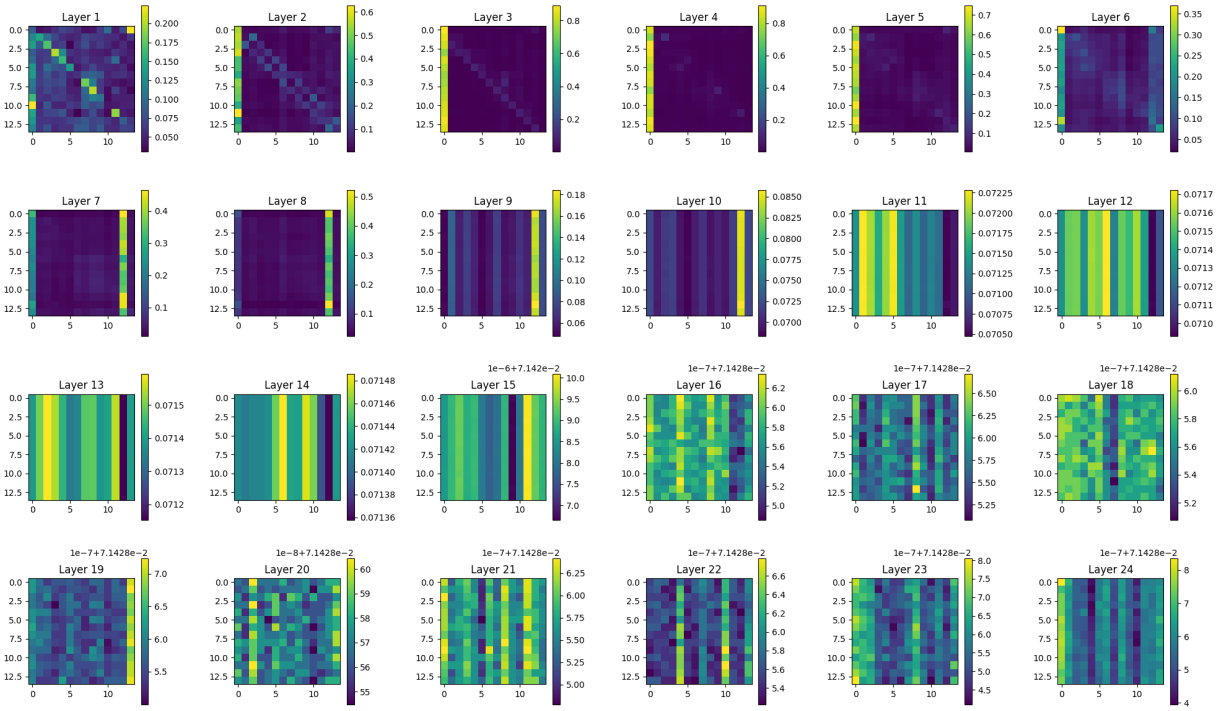

Figure 6: 24-layer attention distribution map from the text encoder.

We employ the k-core to guide deterministic and probabilistic topological enhancement strategies to generate more structured cohesive subgraphs, thereby enriching contrastive learning samples. This approach enables the model to better capture structural and semantic knowledge.

D Supplementary Zero-shot Results

To facilitate a fair comparison, we replace the DeBERTa-large-v3 model in Mirror with RoBERTa-large, re-pretrain and fine-tune it, and then compare the performance under the zero-shot setting. The comparison results between SKIE and Mirror-RoBERTa-Large are shown in Table 12. It can be observed that under the same base model RoBERTa-large, SKIE is still superior to Mirror, demonstrating the effectiveness of our approach.

What's more, we have also included the results of UniNER (Zhou et al., 2024), GoLLIE (Sainz et al., 2024), and GLiNER (Zaratiana et al., 2024) in Table 12 for more comprehensive comparisons. As our pre-training task employs self-supervised contrastive learning, it can support a broader range of unsupervised corpora, but this also inevitably creates a discrepancy with downstream IE tasks. Consequently, the results of SKIE are inferior to these three baselines on 5 NER datasets.

E Error identification

We conduct a comprehensive error identification analysis in NER experiment on ACE05 from three aspects.

Error identification using GSN in SKIE. The sentence is "Thousands more may have been ignored over the last decade. That is the Bush record in policing surgeons, why should we trust him now?" After removing the T-GSN module, SKIE misclassifies "surgeons" as a location, failing to grasp the deep semantic meaning of the sentence. The T-GSN module enhances the model's contextual understanding and reasoning ability through additional relational and cohesive information. Without it, the model might not fully exploit the potential relationships between entities.

Error identification without cohesive subgraphs in SKIE. The sentence is "The ax fell heavily on government and non-profit workers as many state and local governments face severe budget crunches." Without the cohesive subgraph module, SKIE struggles to accurately identify long spans. The cohesive subgraphs, with their multi-level nodes and connections, provide rich semantic and logical structures. By leveraging these, the model can better comprehend complex relationships and concepts in text. Its absence may hinder the model's ability to recognize long or complex entities.

Error identification in other methods but not in SKIE. The sentence is "An automotive tire shop, and someone noticed him, recognized him." While Mirror incorrectly identifies the facility as "shop", SKIE's results are flawless. We attribute this to our method's ability to capture more structural and semantic knowledge, granting it an advantage in long sentences.

F Layers Change Analysis

We conduct layers change of the text encoder analysis using a sentence from ACE05, "Sergeant Chuck Hagel was seriously wounded twice in Vietnam," and input it into our pre-trained model. Figure 6 shows the following observations:

Lower layers (1-5): The attention distribution is relatively even, with minimal gaps in attention scores between words, indicating that the model primarily focuses on basic vocabulary and grammatical structures within the sentence.

Middle layers (6-10): The attention scores between the last word and the preceding text increase, suggesting that the model is beginning to analyze the contextual information of the sentence.

Middle layers (11-15): The attention scores for entities begin to rise significantly, such as "Chuck Hagel" and "Vietnam", indicating that the model is gradually comprehending the roles of these entities within the statement.

Upper layers (16-24): The attention scores for relationships between entities notably increase. For instance, "was wounded" receives a marked boost in attention, demonstrating that the model is enhancing its understanding of the relationships between different entities and the overall semantics within the sentence.